

“ЗАТВЕРДЖУЮ”  
Завідувач кафедри

Білоус Ірина Володимирівна

“\_\_” \_\_\_\_\_ 2021 р.

## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### JAVA ТА C# ТЕХНОЛОГІЇ ПРИКЛАДНОГО ПРОГРАМУВАННЯ (ОК24)

#### Освітня програма «*Інженерія програмного забезпечення*»

Рівень вищої освіти – *перший (бакалаврський)*

Спеціальність 121 – *Інженерія програмного забезпечення*

Мова навчання: *українська*

Статус дисципліни: *обов'язкова*

Форма навчан.	Рік навч.	Сем.	Розподіл годин					Разом	За тиждень		Контр.
			Всього ауд.	Лек	Прак	Лаб.	СРС		Ауд.	СРС	
Денна	3	5	30	16		14	90	120	1,875	5,625	Е
Денна	3	6	40	20		20	110	150	2,86	7,86	Е

Робоча програма Java та C# технології прикладного програмування  
(назва навчальної дисципліни)

для здобувачів вищої освіти галузі знань 12 – *Інформаційні технології*  
спеціальності 121 – *Інженерія програмного забезпечення*

Розробник робочої навчальної програми:

викладач кафедри *інформаційних технологій і програмної інженерії*  
НУ «Чернігівська політехніка»

\_\_\_\_\_ (підпис) (О.О.Дружинін)  
(прізвище та ініціали)

Робочу програму схвалено на засіданні кафедри *інформаційних технологій та програмної інженерії*

Протокол від «31» *серпня* 2021 року № 1

Завідувач кафедри *інформаційних технологій та програмної інженерії*

\_\_\_\_\_ (підпис) (Білоус І.В.)  
(прізвище та ініціали)

## Abstract

### **ESIEIT / SE OK24 – Java & C# technologies of application programming**

#### **2021/2022 Sem. 1**

##### **Course Description**

The modern world of software development is developing very fast. Some new technology or new development approach is invented every day. This course aims not only to inform students about the latest technologies but also to learn how to select the best technology from several. One of the important tasks is to teach students to use different approaches to improve software quality.

The aim of the first part of the course is to familiarize students with the Java platform and with base technologies and technologies for supporting program's life cycle. The second part of the course aims to familiarize students with the techniques for working with databases. The third part of the course aims to familiarize students with web application technologies.

**Contents:** architecture of Java, Java platform, JAXB, JDBC, JUnit, Servlet, JSP, JSTL, Hibernate, Open JPA, unit testing, regression testing, functional testing, design patterns, clean code.

#### **2021/2022 Sem. 2**

##### **Course Description**

The modern world of software development is developing very fast. Some new technology or new development approach are invented every day. This course aims not only to inform students about the latest technologies but also to learn how to select the best technology from several. One of the important tasks is to teach students to use different approaches to improve software quality.

The aim of the first part of the course is to familiarize students with the .NET platform, programming language C # and with imbedded in platform technologies. The second part of the course aims to familiarize students with the techniques for working with databases. The third part of the course aims to familiarize students with web application technologies and software regression testing.

**Contents:** object life cycle architecture of .NET, .NET platform, ADO.NET, ASP.NET, unit testing, regression testing, functional testing, design patterns.

## 1 Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
		денна форма навчання
Кількість кредитів – 9	Галузь знань 12 – Інформаційні технології	Нормативна
Модулів – 2	Спеціальність: 121 – Інженерія програмного забезпечення Освітньо-професійна програма: Інженерія програмного забезпечення	<b>Рік підготовки:</b>
Змістових модулів – 6		3-й
Індивідуальне науково-дослідне завдання – не передбачено		<b>Семестр</b>
Загальна кількість годин – 270		5-й, 6-й
<b>5-й семестр</b> Тижневих годин: аудиторних – 1.875; самостійної і індивідуальної роботи студента – 5,625 <b>6-й семестр</b> Тижневих годин: аудиторних – 2.86; самостійної і індивідуальної роботи студента – 7.86	Рівень вищої освіти: <i>перший (бакалаврський)</i>	<b>Лекції</b>
		5-й семестр - 1 год. 6-й семестр – 1,43 год.
		<b>Лабораторні</b>
		5-й семестр – 0,875 год. 6-й семестр - 1,43 год.
		<b>Самостійна робота</b>
		5-й семестр – 5,625 год. 6-й семестр – 7,86 год.
		<b>Вид контролю:</b>
E		

### Примітка.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи в 5-му семестрі становить 30:90=1:3.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи в 6-му семестрі становить 40:110=1:2,75.

Застосовані скорочення:

JDBC – Java database connectivity

JSP – Java server pages

JSTL – Java standard tag library

ООП – Об'єктно-орієнтовне програмування

- ASP – Active server pages
- ORM Object relation mapping
- ADO – ActiveX data objects

Передумовою для вивчення дисципліни в 5-му семестрі є успішне засвоєння дисциплін *«Веб-технології та веб-дизайн»*, *«Об'єктно-орієнтовне програмування»*, *«Бази даних»* та здобуті такі результати навчання, як знання базових принципів ООП і вміння застосовувати базові принципи ООП під час створення застосунків, вміння створювати застосунки з використанням базових конструкцій і технологій платформи Java. Набуті під час вивчення дисципліни *«Java та C# технології прикладного програмування»* в 5-му семестрі знання та вміння є базовими для подальшого вивчення навчальних дисциплін *«Засоби інтеграції розподілених систем»* та *«Архітектура програмного забезпечення»* та *«Java та C# технології прикладного програмування»*, а також можуть бути застосовані під час підготовки кваліфікаційних робіт бакалаврів та магістрів.

Передумовою для вивчення дисципліни в 6-му семестрі є успішне засвоєння дисципліни *«Бази даних»*, а також курсу *«Java та C# технології прикладного програмування»* за попередній семестр, та здобуті такі результати навчання, як знання базових принципів ООП і вміння застосовувати базові принципи ООП під час створення застосунків, вміння створювати застосунки з використанням базових конструкцій і технологій платформи C#. Набуті під час вивчення дисципліни *«Java та C# технології прикладного програмування»* в 6-му семестрі знання та вміння є базовими для подальшого вивчення навчальних дисциплін *«Скриптові мови програмування»*, *«Системи штучного інтелекту»*, *«Програмування internet-систем»* та також можуть бути застосовані під час підготовки кваліфікаційних робіт бакалаврів та магістрів.

Обов'язковою умовою викладання дисципліни є проведення лабораторного практикуму із застосуванням сучасних персональних комп'ютерів.

## **2 Мета навчальної дисципліни**

Метою викладання навчальної дисципліни *«Java та C# технології прикладного програмування»* є закріплення та розвиток фахових компетентностей бакалавра в галузі знань *12 – Інформаційні технології* із застосування у професійній діяльності сучасних програмних технологій як основи розробки та експлуатації програмних комп'ютерних систем. Зокрема, це:

- ЗК2. Здатність застосовувати знання у практичних ситуаціях;
- ЗК5. Здатність вчитися і оволодівати сучасними знаннями;
- ЗК6. Здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- ЗК13. Здатність працювати в міжнародному контексті;
- ФК17. Здатність розробляти архітектури, модулі та компоненти програмних систем;

- ФК18. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами;
- ФК19. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу;
- ФК20. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки);
- ФК21. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних;
- ФК24. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя;
- ФК25. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення;
- ФК26. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення;
- ФК27. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

### **3 Очікувані результати навчання з дисципліни**

Навчальна дисципліна *“Java та C# технології прикладного програмування”* має допомогти сформуванню наступні програмні результати навчання.

– ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

– ПР04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

– ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

– ПР08. Вміти розробляти людино-машинний інтерфейс.

– ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

– ПР17. Вміти застосовувати методи компонентної розробки програмного і забезпечення.

– ПР18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

– ПР20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.

– ПР25. Мати навички виконання певних ролей в ІТ-проектах будь-якої складності.

У результаті вивчення навчальної дисципліни студент повинен:

**знати:**

1. вимоги до сучасних програмних систем;
2. новітні Java та C# технології та їх специфікації;
3. переваги Java та C# технологій та основні етапи розробки програмних систем та Java та C# компонентів.

**вміти:**

1. застосовувати сучасні Java та C# технології при розробці програмних систем;
2. використовувати загальні системні ресурси в процесі функціонування Java та C# програм.

#### 4 Критерії оцінювання результатів навчання

З тими студентами, які до проведення підсумкового семестрового контролю не встигли виконати всі обов'язкові види робіт та мають підсумкову оцінку менше 20 балів (за шкалою оцінювання), проводяться додаткові індивідуальні заняття, за результатами яких визначається, наскільки глибоко засвоєний матеріал, та чи необхідне повторне вивчення дисципліни.

Дисципліну можна вважати такою, що засвоєна у 5-му семестрі, якщо студент:

**знає:**

- архітектуру платформи Java;
- визначення поняття рефлексії;
- архітектуру JDBC;
- архітектуру та базові класи Hibernate та Open JPA;
- принципи функціонування Servlet та JSP;
- технології тестування програмного забезпечення.

**вміє:**

- застосовувати знання про колекції .NET на практиці;
- створювати додаток, що підключається до бази даних з використанням JDBC і виконує прості CRUD операції;
- створювати додаток, що підключається до бази даних з застосування технології Hibernate і виконує прості CRUD операції.
- створювати та запускати простий web-додаток;
- створювати та запускати простий модульний тест.

Дисципліну можна вважати такою, що засвоєна у 6-му семестрі, якщо студент:

**знає:**

- архітектуру платформи .NET;
- базові конструкції мови C#;
- класифікацію колекцій .NET;

- визначення поняття рефлексії;
- архітектуру ADO.NET;
- архітектуру NHibernate;
- архітектуру ASP.NET;
- технології тестування програмного забезпечення.

**вміє:**

- застосовувати знання про колекції .NET на практиці;
- створювати простий консольний додаток на мові C#;
- створювати та запускати простий web-додаток;
- створювати та запускати простий модульний тест.

## **5 Засоби діагностики результатів навчання**

Засобами оцінювання та методами демонстрування результатів навчання з дисципліни є:

- іспит;
- презентація нових технологій з прикладами їх застосування;
- завдання, які виконуються в навчальній лабораторії;
- інші види індивідуальних та групових завдань.

## **6 Програма навчальної дисципліни**

### **Модуль 1. Java технології**

#### **Змістовний модуль 1. Базові технології Java**

##### **Тема 1. Платформа Java і її основні компоненти**

Архітектура платформи Java. Призначення базових компонентів платформи. Керована динамічна пам'ять, розміщення об'єктів в керованій динамічній пам'яті. Збирання «сміття». Специфікація Java та Java Virtual Machine. Ролі Java розробника в IT-проекті.

##### **Тема 2. Технологія динамічного визначення типів**

Поняття та застосування рефлексії. Клас Class, його значення для рефлексії, поля і методи класу Class. Отримання закритих членів класу. Динамічний виклик методів.

##### **Тема 3. Технології модульного і функціонального тестування**

Загальні принципи модульного і функціонального тестування, інструментальні засоби та бібліотеки підтримки модульного і функціонального тестування.

##### **Тема 4. Технології зв'язування XML даних**

Базові поняття XML, визначення DTD та XSD. Структура XML. Технологія JAXB.



## **Змістовний модуль 2. Технології роботи з базами даних та технології забезпечення застосунків даними**

### **Тема 5. Технологія JDBC**

Архітектура JDBC та призначення кожного з елементів архітектури. Процес з'єднання з базою даних. Додання, видалення та заміна даних в JDBC.

### **Тема 6. Технології об'єктно-реляційного відображення**

Існуючі технології об'єктно-реляційного відображення. Архітектура Hibernate. Вимоги Конфігурація Hibernate та завдання правил відображення з допомогою XML-файлів. Сесія Hibernate, відкриття та закриття сесії, призначення сесії. Опис правил відображення для зв'язків між об'єктами один до одного, один до багатьох, багато до багатьох. Hibernate 4 та написання правил відображення з допомогою анотацій. Відмінності технології Open JPA від Hibernate. Шаблони, що використовуються при застосуванні технологій об'єктно-реляційного відображення.

### **Тема 7. Технології повнотекстового пошуку**

Базові принципи технології Solr. Приклади застосування технології Solr в Java.

### **Тема 8. Технологія розпізнавання мови Sphinx**

Базові принципи технологій розпізнавання мови в Java. Технологія Sphinx. Приклади застосування технології Sphinx.

## **Змістовний модуль 3. Web-технології Java та технології інтегрування застосунків**

### **Тема 9. Java технології Servlet та JSP**

Технологія Servlet та приклади її застосування, технологія JSP як частковий випадок Servlet. Бібліотека тегів JSTL. Теги користувача. AJAX запити з застосуванням технології jQuery.

### **Тема 10. Сервіси Rest та web-сервіси XML**

Відмінності сервісів Rest від web-сервісів XML. Три кроки до Restful-сервісу. Технології, що забезпечують підтримку Rest та приклади застосування цих технологій.

### **Тема 11. Технології інтегрування застосунків**

Базові принципи інтегрування застосунків. Технології RMI та JMS. Приклади інтегрування застосунків з використанням технологій RMI та JMS.

### **Тема 12. Підтримка MVC в web-застосунках**

Огляд технологій підтримки архітектурного шаблону Model-View-Control. Структура бібліотеки Spring MVC та приклади її застосування.

### **Тема 13. Технології автоматизованого збирання проекту**

Огляд технологій автоматизованого збирання проекту. Технології Ant, Maven, Gradle. Інструментальні засоби підтримки технологій Ant, Maven та Gradle. Приклади застосування Ant, Maven та Gradle.

### **Тема 14. Технології обробки великих об'ємів даних**

Технологія Hadoop. Базові принципи технологій обробки великих об'ємів даних. Приклади застосування технології Hadoop.

## **Модуль 2. С# -технології**

### **Змістовний модуль 1. Платформа .NET. Мова програмування С#**

#### **Тема 1. Платформа .NET. Життєвий цикл існування об'єкту**

Архітектура платформи .NET. Призначення базових компонентів платформи. Схема компіляції та виконання програми в .NET. Керована динамічна пам'ять, розміщення об'єктів в керованій динамічній пам'яті. Дії CLR при зустрічі інструкції new. Збирання «сміття». Ролі С# розробника в IT-проекті.

#### **Тема 2. Конструкції мови С#**

Стандартизація мови С# ECMA та ISO/IEC. Базові конструкції мови С#. Ітераційні конструкції. Конструкції розгалужень. Модифікатори параметрів методів. Два види типів в мові С#. Перерахування і інтерфейси. Модифікатори доступу. Абстрактні класи. Статичні поля, методи та конструктори. Узагальнення в мові С#. Делегати та їх застосування.

#### **Тема 3. Колекції С#**

Неузагальнені колекції. Колекції .NET 2.0. Узагальнені колекції List, HashSet, LinkedList, Dictionary, Queue, Stack та їх застосування. Колекції з порозрядною адресацією. Паралельні колекції. Спеціальні колекції.

#### **Тема 4. Рефлексія та атрибути**

Визначення та застосування рефлексії. Метадані в збірці. Клас Type, його значення для рефлексії, поля і методи класу Type. Отримання закритих членів класу. Динамічний виклик методів. Динамічне створення збірки. Атрибути та їх використання.

#### **Тема 5. Серіалізація об'єктів**

Визначення серіалізації об'єктів. Форматери для серіалізації та їх використання. Атрибути для контролю процесу серіалізації. Схема процесу серіалізації. Опис роботи форматера. Налаштування процесу серіалізації. Підтримка версій в серіалізації.

## **Змістовний модуль 2. Технології роботи з базами даних**

### **Тема 6. Технологія ADO.NET**

Архітектура ADO.NET. Зв'язний та незв'язний рівні ADO.NET, умови їх використання. Об'єктна модель ADO.NET. Процес з'єднання з базою даних. Додання, видалення та заміна даних в зв'язному рівні ADO.NET. Додання, видалення, заміна даних в незв'язному рівні ADO.NET. Параметризовані запити. ADO.NET і .NET Core.

### **Тема 7. Технологія LINQ (LINQ to Object, LINQ to XML, LINQ to SQL)**

Архітектура LINQ. Нововведення в мову C# для підтримки LINQ (лямбда вирази, дерева виразів, анонімні типи, розширюючі методи, часткові методи) Конструкції інтегрованої мови запитів. Об'єктна модель LINQ to XML. Конструювання збереження та завантаження дерева XML з допомогою LINQ. Додання, видалення та модифікація вузла в дереві XML. Застосування XPath для роботи з XML. Приклад створення моделі в LINQ to SQL, Entity framework в .NET Core.

### **Тема 8. Технології об'єктно-реляційного відображення (NHibernate, Fluent NHibernate)**

Існуючі технології об'єктно-реляційного відображення. Архітектура NHibernate. Вимоги до класу домену. Конфігурація NHibernate та завдання правил відображення з допомогою XML-файлів. Сесія NHibernate, відкриття та закриття сесії, призначення сесії. Опис правил відображення для зв'язків між об'єктами один до одного, один до багатьох, багато до багатьох. Fluent NHibernate та написання правил відображення з допомогою класів. Запити у NHibernate. Шаблони, що використовуються при застосуванні технологій об'єктно-реляційного відображення.

## **Змістовний модуль 3. Web-технології та Unit-тестування**

### **Тема 9. Технологія ASP.NET**

Взаємодія ASP.NET з Internet Information Server. Життєвий цикл сторінки ASP.NET, етапи життєвого циклу. Шаблони сторінок, .NET Core MVC.

### **Тема 10. WEB-сервіси XML**

Компоненти web-сервіса XML. Архітектура web-сервісів. Способи розробки WEB-сервісів. Приклад WEB-сервісу в C#. Розробка WEB-сервісів шляхом написання WSDL файлів. Механізм пошуку сервісів. Архітектура UDDI.

### **Тема 11. Unit-тестування**

Поняття unit-тестування. Огляд бібліотек Unit-тестування. Базові конструкції Unit-тестування Тестування з допомогою бібліотеки Unit Testing Framework.

## 7 Структура навчальної дисципліни

Назви змістових модулів і тем		Кількість годин для денної/заочної форми навчання							
		Всього		У тому числі					
				Лек.		Лаб.		С.р.	
1	2	3	4	5	6	9	10	11	12
<b>Модуль 1. Java технології</b>									
<b>Змістовний модуль 1. Базові технології Java</b>									
1	Платформа Java і її основні компоненти	6		1					5
2	Технологія динамічного визначення типів	9		2		2			5
3	Технології модульного і функціонального тестування	6		1					5
4	Технології зв'язування XML даних	6		1					5
<b>Разом за змістовний модуль 1</b>		<b>27</b>		<b>5</b>		<b>2</b>			<b>20</b>
<b>Змістовний модуль 2. Технології роботи з базами даних та технології забезпечення застосунків даними</b>									
5	Технологія JDBC	7		2		2			3
6	Технології об'єктно-реляційного відображення	11		3		4			4
7	Технології повнотекстового пошуку	10							10
8	Технологія розпізнавання мови Sphinx	10							10
<b>Разом за змістовний модуль 2</b>		<b>38</b>		<b>5</b>		<b>6</b>			<b>27</b>
<b>Змістовний модуль 3. Web-технології Java та технології автоматизованого збирання проекту</b>									
9	Java технології Servlet та JSP	7		2		2			3
10	Сервіси Rest та web-сервіси XML	9		2		2			5
11	Технології інтегрування застосунків	10							10
12	Підтримка MVC в web-застосунках	10							10
13	Технології автоматизованого збирання проекту	9		2		2			5
14	Технології обробки великих об'ємів даних	10							10
<b>Разом за змістовний модуль 3</b>		<b>55</b>		<b>6</b>		<b>6</b>			<b>43</b>
<b>Усього годин за модуль 1</b>		<b>120</b>		<b>16</b>		<b>14</b>			<b>90</b>
<b>Модуль 1. C# технології</b>									
<b>Змістовний модуль 1. Платформа .NET. Мова програмування C#</b>									
Платформа .NET. Життєвий цикл існування об'єкту		11		1					10

1	2	3	4	5	6	9	10	11	12
	Конструкції мови C#	11		1				10	
	Колекції C#	14		2		2		10	
	Рефлексія та атрибути	16		2		4		10	
	Серіалізація об'єктів	12		2				10	
	<b>Разом за змістовим модулем 1</b>	<b>64</b>		<b>8</b>		<b>6</b>		<b>50</b>	
<b>Змістовий модуль 2. Технології роботи з базами даних</b>									
	Технологія ADO.NET	16		2		4		10	
	Технологія LINQ (LINQ to Object, LINQ to XML, LINQ to SQL)	12		2				10	
	Технології об'єктно-реляційного відображення (NHibernate, Fluent NHibernate)	16		2		4		10	
	<b>Разом за змістовим модулем 2</b>	<b>44</b>		<b>6</b>		<b>8</b>		<b>30</b>	
<b>Змістовий модуль 3. WEB-технології та Unit-тестування</b>									
	Технологія ASP.NET	16		2		4		10	
	Web-сервіси XML	12		2				10	
	Unit-тестування	14		2		2		10	
	<b>Разом за змістовим модулем 3</b>	<b>42</b>		<b>6</b>		<b>6</b>		<b>30</b>	
	<b>Усього годин за модуль 2</b>	<b>150</b>		<b>20</b>		<b>20</b>		<b>110</b>	
	<b>Усього годин за дисципліну</b>	<b>270</b>		<b>36</b>		<b>34</b>		<b>200</b>	

## 8 Теми лабораторних занять

№ з/п	Назва теми	Кількість годин (д/з)
<b>Модуль 1. Java технології</b>		
1	Вивчення динамічного визначення типів	2
2	Доступ до СУРБД з використанням пакету JDBC	2
3	Вивчення об'єктно-реляційного відображення	4
4	Вивчення Java технологій Servlet і JSP	2
5	Створення web-сервісу в IDE Eclipse	2
6	Вивчення технологій автоматизованого збирання проекту	2
<b>Разом за модуль 1</b>		<b>14</b>
<b>Модуль 2. C# технології</b>		
1	Ознайомлення з різновидами, призначенням та функціональними можливостями колекцій C#	2
2	Ознайомлення з призначенням та функціональними можливостями механізму рефлексії в C#	4
3	Ознайомлення з функціональними можливостями технології ADO.NET	4
4	Ознайомлення з призначенням і функціональними можливостями технології Fluent NHibernate	4
5	Ознайомлення з призначенням та функціональними можливостями	4

№ з/п	Назва теми	Кількість годин (д/з)
	технології ASP.NET	
6	Ознайомлення з призначенням і функціональними можливостями бібліотеки unit-тестування Unit Testing Framework	2
<b>Разом за модуль 2</b>		<b>20</b>
<b>Разом</b>		<b>34</b>

## 9 Самостійна робота

№ з/п	Назва теми	Кількість годин (д/з)
<b>Модуль 1. Java технології</b>		
1	Платформа Java і її основні компоненти	5
2	Технологія динамічного визначення типів	5
3	Технології модульного і функціонального тестування	5
4	Технології зв'язування XML даних	5
5	Технологія JDBC	3
6	Технології об'єктно-реляційного відображення	4
7	Технології повнотекстового пошуку	10
8	Технологія розпізнавання мови Sphinx	10
9	Java технології Servlet та JSP	3
10	Сервіси Rest та web-сервіси XML	5
11	Технології інтегрування застосунків	10
12	Підтримка MVC в web-застосунках	10
13	Технології автоматизованого збирання проекту	5
14	Технології обробки великих об'ємів даних	10
<b>Разом за модуль 1</b>		<b>90</b>
<b>Модуль 2. C# технології</b>		
1	Платформа .NET. Життєвий цикл існування об'єкту	10
2	Конструкції мови C#	10
3	Колекції C#	10
4	Рефлексія та атрибути	10
5	Серіалізація об'єктів	10
6	Технологія ADO.NET	10
7	Технологія LINQ (LINQ to Object, LINQ to XML, LINQ to SQL)	10
8	Технології об'єктно-реляційного відображення (NHibernate, Fluent NHibernate)	10
9	Технологія ASP.NET	10
10	Web-сервіси XML	10
11	Unit-тестування	10
<b>Разом за модуль 2</b>		<b>110</b>
<b>Разом</b>		<b>200</b>

## 10 Індивідуальні завдання

Навчальним планом не передбачено виконання індивідуальних завдань.

## 11 Методи контролю

Оцінювання знань студентів здійснюється відповідно до [«Положення про поточне та підсумкове оцінювання знань здобувачів вищої освіти Національного університету «Чернігівська політехніка»](#).

Політика дотримання академічної доброчесності ґрунтується на [«Кодексі академічної доброчесності Національного університету «Чернігівська політехніка»](#). Жодні форми порушення академічної доброчесності не толеруються.

З дисципліни студент може набрати до 60% підсумкової оцінки за виконання всіх видів робіт, що виконуються протягом семестру і до 40% підсумкової оцінки – на іспиті.

Виконання та особистий захист усіх лабораторних робіт, зазначених у робочій навчальній програмі з дисципліни, є обов'язковим. Поточний контроль проводиться шляхом написання студентами змістовних модульних тестів та опитування студентів під час захисту лабораторних робіт. Допускається отримання студентами додаткових балів за підготовку презентації нових Java або C# технологій, що не були розглянуті в рамках курсу лекцій. За такий вид роботи студент може отримати від 1 до 15 додаткових балів за кожен з модулів. Студенти не отримують додаткові бали за відвідування лекцій, але кожна відсутність на лекції з неповажної причини віднімає від загальної кількості балів, отриманої під час модуля, 2 бали.

Результати поточного контролю за відповідний модуль оприлюднюються викладачем на наступному аудиторному занятті. Бали, які набрані студентом під час підсумкових модульних тестів і бали, отримані за захист лабораторних робіт, складають оцінку поточного контролю.

Семестровий контроль у вигляді іспиту проводиться під час сесії з трьома запитаннями: двома теоретичними (по 10 балів максимум за кожне) та одним практичним (20 балів максимум). Оцінка за результатами вивчення дисципліни формується шляхом додавання підсумкових результатів поточного контролю до оцінки, отриманої під час екзамену. Ті студенти, які не виконали всі обов'язкові види робіт та за результатами роботи в семестрі набрали менше 20 балів, мають пройти повторний курс вивчення дисципліни.

Якщо відповідь повна і зміст відповіді студента повністю відповідає сутності поставленого запитання, можна отримати від 33 до 40 балів. В тому випадку, коли студент виконує всі завдання без грубих помилок, можна отримати від 24 до 32 балів. Якщо при виконанні завдань студент допускає грубі помилки, і всі запитання вирішені менш, ніж на половину, можна отримати від 17 до 24 балів. За невиконання хоча б одного завдання, не можна отримати більше 16 балів.

Складання екзамену є обов'язковим елементом підсумкового контролю знань для студентів, які претендують на оцінку «добре» або «відмінно». Якщо студент виконав всі види робіт протягом семестру та набрав 60% підсумкової оцінки (тобто «задовільно»), то він, за бажанням, може залишити набрану кількість балів як підсумкову оцінку і не складати іспит.

У випадку, якщо здобувач протягом семестру не виконав в повному обсязі передбачених робочою програмою всіх видів навчальної роботи, має невідпрацьовані лабораторні роботи або не набрав мінімально необхідну кількість балів, він не допускається до складання іспиту під час семестрового контролю, але має право ліквідувати академічну заборгованість у порядку, передбаченому [«Положенням про поточне та підсумкове оцінювання знань здобувачів вищої освіти Національного університету «Чернігівська політехніка»](#).

Повторне складання іспиту з метою підвищення позитивної оцінки не дозволяється.

## 12 Розподіл балів, які отримують студенти

### Поточний контроль за модулями

Модуль за тематичним планом дисципліни та форма контролю	Кількість балів
<b>Модуль 1. Java технології</b>	<b>0... 60</b>
<b>Змістовний модуль 1. Базові технології Java</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №1	0... 5
2 Виконання та захист лабораторної роботи №2	0... 5
3 Написання змістовного модуля №1	0... 5
4 Підготовка презентації (додаткові бали)	0... 5
<b>Змістовний модуль 2. Технології роботи з базами даних та технології забезпечення застосунків даними</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №3	0... 5
2 Виконання та захист лабораторної роботи №4	0... 5
3 Написання змістовного модуля №2	0... 5
4 Підготовка презентації (додаткові бали)	0... 5
<b>Змістовний модуль 3. Web-технології Java та технології автоматизованого збирання проекту</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №5	0... 5
2 Виконання та захист лабораторної роботи №6	0... 5
3 Написання змістовного модуля №3	0... 5
5 Підготовка презентації (додаткові бали)	0... 5
<b>Модуль 2. C# технології</b>	<b>0... 60</b>
<b>Змістовний модуль 1. Платформа .NET. Мова програмування C#</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №1	0... 5
2 Виконання та захист лабораторної роботи №2	0... 5
3 Написання змістовного модуля №1	0... 5
4 Підготовка презентації (додаткові бали)	0... 5



<b>Змістовий модуль 2. Технології роботи з базами даних</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №3	0... 5
2 Виконання та захист лабораторної роботи №4	0... 5
3 Написання змістовного модуля №2	0... 5
4 Підготовка презентації (додаткові бали)	0... 5
<b>Змістовий модуль 3. Web-технології та Unit-тестування</b>	<b>0... 20</b>
1 Виконання та захист лабораторної роботи №5	0... 5
2 Виконання та захист лабораторної роботи №6	0... 5
3 Написання змістовного модуля №3	0... 5
5 Підготовка презентації (додаткові бали)	0... 5

### Підсумковий контроль

<b>Структура підсумкового контролю</b>	<b>Кількість балів</b>
1 Теоретичне питання №1	0... 10
2 Теоретичне питання №2	0... 10
3 Практичне завдання	0... 20
4 Сумарне значення результатів по модулям	0... 60
<b>Зважена семестрова оцінка</b>	<b>0... 100</b>

### Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсової роботи	для заліку
90 – 100	<b>A</b>	відмінно	зараховано
82-89	<b>B</b>	добре	
74-81	<b>C</b>		
64-73	<b>D</b>	задовільно	
60-63	<b>E</b>		
35-59	<b>FX</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

### 13 Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

Лекційний матеріал подається у вигляді презентацій за допомогою медіа-проектора або виведення на монітори робочих станцій. Під час лекцій

аналізуються проблемні ситуації, організується зворотний зв'язок з аудиторією шляхом формулювання запитань і стислих відповідей з обох сторін.

#### **14 Методичне забезпечення**

1. Java технології прикладного програмування. Методичні вказівки до виконання лабораторних, розрахунково-графічних робіт та самостійної роботи здобувачів вищої освіти з дисциплін «Java та C# технології прикладного програмування» та «Технології прикладного програмування» для здобувачів вищої освіти першого (бакалаврського) рівня вищої освіти спеціальностей 121 – «Інженерія програмного забезпечення» та 123 – «Комп'ютерна інженерія» / Укл.: Задорожній А.О., Богдан І.В., Войцеховська М.М., Дружинін О.О. – Чернігів: ЧНТУ, 2020. – 146 с.
2. Литвинов В. В., Казимир В.В., Пріла О. А., Харченко М. В., Задорожній А. О. Лабораторний практикум з об'єктно-орієнтованих технологій розробки прикладних програм. Навчальний посібник.–Чернігів: видавництво Чернігівського державного інституту економіки і управління, 2014.- 364 с.
3. C# технології для створення web-застосунків. Методичні вказівки до лабораторних робіт з дисципліни «Java та C# технології прикладного програмування» для студентів спеціальності 121 – «Інженерія програмного забезпечення» / Укл. А.О. Задорожній, І.В. Білоус, М.М. Войцеховська. – Чернігів: ЧНТУ, 2020, укр. мовою.

#### **15 Рекомендована література**

##### **Базова**

1. Shildt Herbert. Java Complete guide. Eleventh edition. – McGraw-Hill Education, 2018. – 1248 p.
2. Kurniawan B. Servlet & JSP: A Tutorial, Second Edition. Brainy Software, 2015. - 630 p.
3. Benjamin Muschko. Gradle in Action. Manning, 2014.- 395 p.
4. Konda M. Just Hibernate: A Lightweight Introduction to the Hibernate Framework 1st Edition. O'Reilly, 2014.-140 p.
5. Griffiths I., Adams M., Liberty J. Programming C# 4.0, 6th Edition.: O'Reilly, 2010. – 856 p.
6. Roy Osherove. The Art of Unit Testing: with examples in C#, 2nd Edition.: Manning Publications, 2013.-296 p.
7. Mark J. Price. C# 8.0 and .NET Core 3.0 – Modern Cross-Platform Development: Build applications with C#, .NET Core, Entity Framework Core, ASP.NET Core, and ML.NET using Visual Studio Code, 4th Edition.: Packt, 2019.- 818 p.
8. Harrison Ferrone. Learning C# by Developing Games with Unity 2019: Code in C# and build 3D games with Unity, 4th Edition.: Packt, 2019.- 342 p.
9. Liljas G., Zaytsev A., Dentler J. NHibernate 4.x Cookbook - Second Edition. Packt Publishing, 2017. – 448 p.

**Допоміжна**

10. Robert C Martin. Clean Code : A Handbook of Agile Software Craftsmanship. Prentice Hall, 2008. - 464 p.
11. Bloch Joshua. Effective Java. Third Edition. – Addison-Wesley, 2018. – 413 p.

**16 Інформаційні ресурси**

1. Система дистанційного навчання НУ «Чернігівська політехніка». Курс: Java та C# технології прикладного програмування. – [Електронний ресурс]. – Режим доступу : <https://eln.stu.cn.ua/enrol/index.php?id=532>
2. Java Documentation [Electronic resource]. – Access mode: <https://docs.oracle.com/en/java/>
3. The Java Language Specification, Java SE 16 Edition [Electronic resource]. – Access mode: <https://docs.oracle.com/javase/specs/jls/se16/jls16.pdf>
4. The Java Virtual Machine Specification, Java SE 16 Edition [Electronic resource]. – Access mode: <https://docs.oracle.com/javase/specs/jvms/se16/jvms16.pdf>
5. .NET Documentation [Electronic resource]. – Access mode: <https://docs.microsoft.com/en-us/dotnet/>
6. ECMA-334 C# Language Specification [Electronic resource]. – Access mode: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-334.pdf>
7. ISO/IEC 23270:2018 Information technology -- C# Language Specification [Electronic resource]. — Access mode: [https://standards.iso.org/ittf/PubliclyAvailableStandards/c075178\\_ISO\\_IEC\\_23270\\_2018.zip](https://standards.iso.org/ittf/PubliclyAvailableStandards/c075178_ISO_IEC_23270_2018.zip)